

# Dialler REST API - Dialler Contact Functions

- [Retrieve Dialler Contacts](#)
- [Retrieve a specific Dialler Contact by ID](#)
- [Retrieve Dialler Contacts by User Data](#)
- [Import a Dialler Contact](#)
- [Update a Dialler Contact](#)
- [Delete a Dialler Contact by ID](#)
- [Delete a Dialler Contact by UserData](#)
- [Lock a Dialler Contact by ID](#)
- [Lock a Dialler Contact by UserData](#)
- [Unlock a Dialler Contact by ID](#)
- [Unlock a Dialler Contact by UserData](#)
- [Reset a Dialler Contact by ID](#)
- [Reset a Dialler Contact by UserData](#)
- [Retrieve Dialler Contact History by ID](#)
- [Retrieve Dialler Contact History by UserData](#)

## Retrieve Dialler Contacts

This function retrieves all dialler contacts from the client account or from a specific dialler campaign.

### URL

```
GET {baseUrl}/dialler/contacts?diallerCampaignsID={diallerCampaignsID}
```

### Parameters

Parameter	Where	Type	Data
diallerCampaignsID	Query	Integer	The ID of the Dialler Campaign to retrieve the contacts from.  If not passed, all contacts in the client account will be retrieved.  Caution - this function will return a lot of data if the campaigns contain lots of contacts.

### Responses

If the request is performed, 200 OK is returned. See [Return Codes](#) for further possible status codes.

### Returned Data (200 OK)

A JSON Structure containing the following fields:

Field	Type	Data
-------	------	------

filter	String	The applied filter. This is reserved for future use.
limitStart	Integer	The first record index. Currently 0, reserved for future use.
limitPageSize	Integer	The number of records. -1 indicates all records. Reserved for future use.
records	DiallerCampaigns []	An array of dialler campaigns.
count	Integer	The number of dialler campaigns returned.
orderBy	String	The field used to sort the dialler campaigns.
ascending	Boolean	Whether the array of campaigns is sorted ascending or descending.
status	Integer	The return code (also provided by the http status code).

## Example

Note, the dialler contact array has been shortened for readability.

```
{
  "filter": "",
  "limitStart": 0,
  "limitPageSize": -1,
  "records": [
    {
      "bLastResultMailbox": null,
      "bPhoneNumber3Invalid": null,
      "bPhoneNumber4Invalid": null,
      "Company": "Musterfirma GmbH",
      "ExternalSystemLink": "https://crm.mycrm.com?uid=123456789",
      "Email": null,
      "LastAcidAgentEndReasonsID": null,
      "Address": "Kunde",
      "bPhoneNumber6Invalid": null,
      "PhoneNumber2LastAcidAgentEndReasonsID": null,
      "dtNextTry": null,
      "bPhoneNumber5Invalid": null,
      "bPhoneNumber1Invalid": null,
      "bPhoneNumber2Invalid": null,
      "Name": "Mustermann",
      ...
      "nPhoneNumber4LastCause": null,
      "Model": null,
      "PhoneNumber3LastAcidAgentEndReasonsID": null,
      "Tag": null
    }
  ],
  "count": 147,
  "orderBy": "Name",
  "ascending": true,
  "status": 200
}
```

## Retrieve a specific Dialler Contact by ID

This function retrieves a specific dialler contact by ID.

### URL

```
GET {baseUrl}/dialler/contacts/{diallerContactsID}
```

### Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the Dialler Contact.

## Responses

If the request is performed, 200 OK is returned. See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the returned contact.

Field	Type	Data
records	DiallerContacts []	An array of dialler contacts, containing one contact.
count	Integer	1
status	Integer	The return code (also provided by the http status code).

## Example

Note, the dialler campaign array has been shortened for readability.

```
{
  "records": [
    {
      "bLastResultMailbox": null,
      "bPhoneNumber3Invalid": null,
      "bPhoneNumber4Invalid": null,
      "Company": "Musterfirma GmbH",
      "ExternalSystemLink": null,
      "Email": null,
      "LastAcidAgentEndReasonsID": null,
      "Address": null,
      "bPhoneNumber6Invalid": null,
      "PhoneNumber2LastAcidAgentEndReasonsID": null,
      "dtNextTry": null,
      "bPhoneNumber5Invalid": null,
      "bPhoneNumber1Invalid": null,
      "bPhoneNumber2Invalid": null,
      "Name": "Mustermann",
      "nTriesPhoneNumber5": 0,
      "nTriesPhoneNumber4": 0,
      "nTriesPhoneNumber6": 0,
      ...
      "nPhoneNumber6LastCause": null,
      "Model": null,
      "PhoneNumber3LastAcidAgentEndReasonsID": null,
      "Tag": null
    }
  ],
  "count": 1,
  "status": 200
}
```

## Retrieve Dialler Contacts by User Data

This function retrieves the dialler contacts given the UserData value to search for.

### URL

```
GET {baseUrl}/dialler/contacts/contact?userData={userData}
```

### Parameters

Parameter	Where	Type	Data
-----------	-------	------	------

userData	Query	String	The userData of the dialler contacts to search for.
----------	-------	--------	---

## Responses

If the request is performed, 200 OK is returned. See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the returned contacts.

Field	Type	Data
records	DiallerContacts[]	An array of dialler contacts.
count	Integer	The number of records returned.
status	Integer	The return code (also provided by the http status code).

## Example

Note, the dialler contact data has been shortened for readability.

```
{
  "records": [
    {
      "bLastResultMailbox": null,
      "bPhoneNumber3Invalid": null,
      "bPhoneNumber4Invalid": null,
      "Company": "Musterfirma GmbH",
      "ExternalSystemLink": null,
      "Email": null,
      "LastAcdAgentEndReasonsID": null,
      "Address": null,
      "bPhoneNumber6Invalid": null,
      "PhoneNumber2LastAcdAgentEndReasonsID": null,
      "dtNextTry": null,
      "bPhoneNumber5Invalid": null,
      "bPhoneNumber1Invalid": null,
      "bPhoneNumber2Invalid": null,
      "Name": "Mustermann",
      "nTriesPhoneNumber5": 0,
      "nTriesPhoneNumber4": 0,
      "nTriesPhoneNumber6": 0,
      ...
      "nPhoneNumber6LastCause": null,
      "Model": null,
      "PhoneNumber3LastAcdAgentEndReasonsID": null,
      "Tag": null
    }
  ],
  "count": 1,
  "status": 200
}
```

## Import a Dialler Contact

This function imports a dialler contact into a specific dialler campaign specified by either the ID or name of the Dialler Campaign.

### URL

POST {baseUrl}/dialler/contacts/import?overwriteDuplicates={overwriteDuplicates}

### Parameters

Parameter	Where	Type	Data
-----------	-------	------	------

overwriteDuplicates	Query	Boolean	Whether to overwrite duplicate contacts.  If this is true, existing contact data will be overwritten when a match on the UserData field is detected.  If this is false, the new contact will be created whether a matching record is found or not.
---------------------	-------	---------	--

## Post Data

A JSON structure containing the contact to import.

Note that the following fields are mandatory:

- Name
- DiallerCampaignsID **OR** the DiallerCampaignsName
- At least one of
  - PhoneNumber1
  - ...
  - PhoneNumber6

The other fields may be provided or left empty.

**Example (all fields, provide DiallerCampaignsID):**



```

{
  "DiallerCampaignsID": 538,
  "RandomSortOrder": null,
  "UserData": "testUserData",
  "Client": "Client",
  "Service": "Service",
  "ReferenceNumber": "ReferenceNumber",
  "Name" : "Name",
  "FirstName" : "FirstName",
  "Salutation" : "Salutation",
  "Title" : "Title",
  "Company" : "Company",
  "Address" : "Address",
  "PostalCode" : "PostalCode",
  "City" : "City",
  "Country" : "Country",
  "AccountOwner" : "AccountOwner",
  "AccountBank" : "AccountBank",
  "AccountNationalBankCode" : "AccountNationalBankCode",
  "AccountBIC" : "AccountBIC",
  "AccountNumber" : "AccountNumber",
  "AccountIBAN" : "AccountIBAN",
  "Amount": 0.0,
  "PhoneNumber1" : "PhoneNumber1",
  "PhoneNumber2" : "PhoneNumber2",
  "PhoneNumber3" : "PhoneNumber3",
  "PhoneNumber4" : "PhoneNumber4",
  "PhoneNumber5" : "PhoneNumber5",
  "PhoneNumber6" : "PhoneNumber6",
  "FaxNumber" : "FaxNumber",
  "Email" : "Email",
  "Website" : "Website",
  "Comment" : "Comment",
  "ReservedUsersUID" : "ReservedUsersUID",
  "FollowUpUsersUID" : "FollowUpUsersUID",
  "FollowUpDateTime" : null,
  "dtBeginDate" : "2020-01-01T00:00:00",
  "dtEndDate" : "2020-03-01T00:00:00",
  "Manufacturer" : "Manufacturer",
  "Model" : "Model",
  "Tag": "Tag",
  "ExternalSystemLink": "http://test.external.link/"
}

```

**Example (shortened, provide DiallerCampaignsName):**

```
{
  "DiallerCampaignsName": "My Campaign",
  "RandomSortOrder": null,
  "UserData": "testUserData",
  "Client": "Client",
  "Service": "Service",
  "ReferenceNumber": "ReferenceNumber",
  "Name" : "Name",
  ...
  "ExternalSystemLink": "http://test.external.link/"
}
```

## Responses

If the request is performed, 200 OK is returned, returning the imported or already existing contact if it was a duplicate.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the imported contact.

Field	Type	Data
records	DiallerContacts[]	An array of dialler contacts.
count	Integer	The number of records returned.
status	Integer	The return code (also provided by the http status code).

## Example

Note, the dialler contact data has been shortened for readability.

```
{
  "records": [
    {
      "bLastResultMailbox": null,
      "bPhoneNumber3Invalid": null,
      "bPhoneNumber4Invalid": null,
      "Company": "Company",
      "ExternalSystemLink": "http://test.external.link/",
      "Email": "Email",
      "LastAcidAgentEndReasonsID": null,
      ...
      "PhoneNumber3LastAcidAgentEndReasonsID": null,
      "Tag": "Tag"
    }
  ],
  "count": 1,
  "status": 200
}
```

## Update a Dialler Contact

This function updates an existing dialler contact into a specific dialler campaign specified by either the ID or UserData, depending on what is supplied in the JSON contact data.

- If the ID is specified, then this function can be used to update the UserData field, if specified.
- If only UserData is specified, then the contact must be found by UserData. Note, that it is not possible to determine which contact exactly will be updated if more than one contact exists with the same UserData field value.
- Only the fields specified in the JSON data structure for the contact will be updated.

### URL

```
POST {baseUrl}/dialler/contacts/update
```

### Post Data

A JSON structure containing the contact to update.

Note that the following fields are mandatory:

- ID
- **OR**
- UserData

The other fields may be provided or left empty.

**Example (all fields which can be updated, provide ID):**

```
{
  "ID": 123456,
  "UserData": "newUserData",
  "UserData2": "newUserData2",
  "UserData3": "newUserData3",
  "Client": "NewClient",
  "Service": "NewService",
  "ReferenceNumber": "NewReferenceNumber",
  "Name" : "NewName",
  "FirstName" : "NewFirstName",
  "Salutation" : "NewSalutation",
  "Title" : "NewTitle",
  "Company" : "NewCompany",
  "Address" : "NewAddress",
  "PostalCode" : "NewPostCode",
  "City" : "NewCity",
  "Country" : "NewCountry",
  "AccountOwner" : "NewAccountOwner",
  "AccountBank" : "NewAccountBank",
  "AccountNationalBankCode" : "NewAccountNationalBankCode",
  "AccountBIC" : "NewAccountBIC",
  "AccountNumber" : "NewAccountNumber",
  "AccountIBAN" : "NewAccountIBAN",
  "Amount": 999.99,
  "PhoneNumber1" : "NewPhoneNumber1",
  "PhoneNumber2" : "NewPhoneNumber2",
  "PhoneNumber3" : "NewPhoneNumber3",
  "PhoneNumber4" : "NewPhoneNumber4",
  "PhoneNumber5" : "NewPhoneNumber5",
  "PhoneNumber6" : "NewPhoneNumber6",
  "FaxNumber" : "NewFaxNumber",
  "Email" : "NewEmail",
  "Website" : "NewWebsite",
  "Comment" : "NewComment",
  "Region" : "NewRegion",
  "dtBeginDate" : "2030-01-01T00:00:00",
  "dtEndDate" : "2030-03-01T00:00:00",
  "Manufacturer" : "NewManufacturer",
  "Model" : "NewModel",
  "Tag": "NewTag",
  "ExternalSystemLink": "http://test.external.link/NewURL"
}
```

**Example (provide UserData, partial update only):**

```
{
  "UserData": "MyUserData",
  "Name" : "CorrectedName",
  "FirstName" : "CorrectedFirstName"
}
```

## Responses

If the request is performed, 200 OK is returned, and the updated contact is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the updated contact.

Field	Type	Data
records	DiallerContacts[]	An array containing the updated contact.
count	Integer	The number of records returned (will always be 1).
status	Integer	The return code (also provided by the http status code).

## Example

Note, the dialler contact data has been shortened for readability.

```
{
  "records": [
    {
      "bLastResultMailbox": false,
      "bPhoneNumber3Invalid": false,
      "bPhoneNumber4Invalid": false,
      "Company": "Company",
      "ExternalSystemLink": "http://test.external.link/",
      "Email": "Email",
      "LastAcidAgentEndReasonsID": null,
      ...
      "PhoneNumber3LastAcidAgentEndReasonsID": null,
      "Tag": "Tag"
    }
  ],
  "count": 1,
  "status": 200
}
```

# Delete a Dialler Contact by ID

This function deletes a dialler contact including all related data by ID.

## URL

```
DELETE {baseUrl}/dialler/contacts/contact/{diallerContactsID}
```

## Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the dialler contact to delete.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Delete a Dialler Contact by UserData

This function deletes a dialler contact including all related data by providing the UserData for the record to delete.

Note, if more than one record contains the same user data, the first one found in the database will be deleted. It is not possible to determine which contact this will actually be.

## URL

```
DELETE /dialler/contacts/contact?userData={userData}
```

## Parameters

Parameter	Where	Type	Data
userData	Query	String	The UserData of the dialler contact to delete.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Lock a Dialler Contact by ID

This function locks a dialler contact by ID.

## URL

```
PATCH {baseUrl}/dialler/contacts/contact/{diallerContactsID}/lock
```

## Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the dialler contact to lock.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Lock a Dialler Contact by UserData

This function locks a dialler contact by UserData.

Note, if more than one record contains the same user data, the first one found in the database will be locked. It is not possible to determine which contact this will actually be.

## URL

```
PATCH {baseUrl}/dialler/contacts/contact/lock?userData={userData}
```

## Parameters

Parameter	Where	Type	Data
userData	Query	String	The userData of the contact to lock.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Unlock a Dialler Contact by ID

This function unlocks a dialler contact by ID.

## URL



```
PATCH {baseUrl}/dialler/contacts/contact/{diallerContactsID}/unlock
```

## Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the dialler contact to lock.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Unlock a Dialler Contact by UserData

This function unlocks a dialler contact by UserData.

Note, if more than one record contains the same user data, the first one found in the database will be unlocked. It is not possible to determine which contact this will actually be.

## URL

```
PATCH {baseUrl}/dialler/contacts/contact/unlock?userData={userData}
```

## Parameters

Parameter	Where	Type	Data
userData	Query	String	The userData of the contact to lock.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Reset a Dialler Contact by ID

This function reset a dialler contact by ID. Resetting a dialler contact sets all call try counters etc. to 0 so the contact is as if it was just imported to the dialler.

## URL

```
PATCH {baseUrl}/dialler/contacts/contact/{diallerContactsID}/reset
```

## Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the dialler contact to reset.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Reset a Dialler Contact by UserData

This function resets a dialler contact by UserData. Resetting a dialler contact sets all call try counters etc. to 0 so the contact is as if it was just imported to the dialler.

Note, if more than one record contains the same user data, the first one found in the database will be reset. It is not possible to determine which contact this will actually be.

## URL

```
PATCH {baseUrl}/dialler/contacts/contact/reset?userData={userData}
```

## Parameters

Parameter	Where	Type	Data
userData	Query	String	The userData of the contact to reset.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the status code.

Field	Type	Data
status	Integer	The return code (also provided by the http status code).

# Retrieve Dialler Contact History by ID

This function retrieves all history (calls and recorded result codes) for a dialler contact, given the dialler contact ID.

## URL

```
GET {baseUrl}/dialler/contacts/contact/{diallerContactsID}/history
```

## Parameters

Parameter	Where	Type	Data
diallerContactsID	URL	Integer	The ID of the dialler contact to retrieve the data from.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the imported contact.

Field	Type	Data
records	DiallerContactsHistory[]	An array of dialler contacts history.
count	Integer	The number of records returned.
status	Integer	The return code (also provided by the http status code).

## Example:

```

{
  "records": [
    {
      "bOutbound": true,
      "dtCallEnd": "2020-11-11T15:31:47.070",
      "UserData": "Contact10",
      "ServiceNumbersID": 99,
      "DiallerCampaignsID": 551,
      "ANumber_CountryCodesID": 49,
      "BNumber_CountryCodesID": 49,
      "DiallerContactsID": 215220,
      "ANumber_AreaCodesID": 4645,
      "CONNRES": 1,
      "BNumber_AreaCodesID": 4645,
      "bCallerHangup": true,
      "ANumber": "49894614950703",
      "dtCallConnect": "2020-11-11T15:31:44.117",
      "dtCreatedDiallerContactsHistory": "2020-11-11T15:32:07.050",
      "CauseCodesID": 31,
      "dtCallAlert": "2020-11-11T15:31:43.190",
      "PhoneNumberOffset": 1,
      "DiallerCampaignsResultCodesID": 2811,
      "StatisticsPartAID": 223402,
      "UsersID": 3,
      "BNumber": "49894614950665",
      "UserData2": null,
      "ClientsID": 1,
      "UserData3": null,
      "dtCallStart": "2020-11-11T15:31:41.649",
      "bTestCall": false,
      "PhoneNumber": "49894614950703",
      "DiallerContactsHistoryID": 1136,
      "DiallerResultCodesTypesID": 3
    }
  ],
  "count": 1,
  "status": 200
}

```

## Retrieve Dialler Contact History by UserData

This function retrieves all history (calls and recorded result codes) for all dialler contacts with a particular UserData field value.

### URL

```
GET {baseUrl}/dialler/contacts/contact/history?userData={userData}
```

## Parameters

Parameter	Where	Type	Data
userData	Query	String	The userData of the dialler contacts to search for.

## Responses

If the request is performed, **200 OK** is returned.

If the contact does not exist, **404 Not Found** is returned.

See [Return Codes](#) for further possible status codes.

## Returned Data (200 OK)

A JSON Structure containing the imported contact.

Field	Type	Data
records	DiallerContactsHistory[]	An array of dialler contacts history.
count	Integer	The number of records returned.
status	Integer	The return code (also provided by the http status code).